52) Consider the following ANSI C code segment :

```
z =x+3 +y->f1 +y->f2;
for (i=0; i< 200; i=i+2){
    if(z>i){
        p=p+x+3;
        q=q+y->f1;
    } else {
        p =p+y->f2;
        q=q+x+3;
    }
}
```

Assume that variable y points to a struct (allocated on the heap) containing two fields f1 and f2 and the local variables x, y, z, p, q and i are alloted registers. Common sub-expression elimination (CSE) optimization is applied on the code. The number of addition and dereference operations (of the form y→f1 or y→f2) in the optimized code, respectively are:

a) 403 & 102 b) 203 & 2 c) 303 & 102 d) 303 & 2

Sol

```
t1 = x+3;   -> addition
t2 = y->f1; -> dereference
t3 = y->f2; -> dereference
z = t1+t2+t3; -> 2 additions

for( ){
        } => 2 additions, 100 times. and 2 dereference operations.
}
```

$100 * 2 + 100 + 3 = 303$

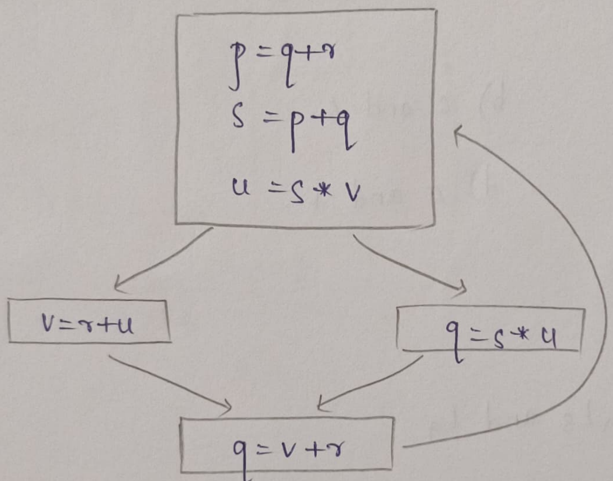
53) A variable x is said to be live at a statement S_i in a program if the following

1. three conditions hold simultaneously.

2. There exists a statement S_j that uses x.

3. There is a path from S_i to S_j in the flow graph corresponding to the program.

The path has no intervening assignment to x including at S_i & S_j



The variables which are live both at the statement in basic block 2 and the statement in basic block 3 of above control flow graph are :

a) p, s, u b) r, s, u [✓ c) r, u] d) q, v

Sol Variables live at Basic block 2 : {r, u}

Variables live at Basic block 3 : {r, s, u}

Variable live at Both basic blocks 2 & 3 : {r, u}

54) Which of the following options correctly specify the number of basic blocks and the number of instructions in the largest basic

block, respectively? Consider the following pseudo-code

```
L1 : t1 = -1
L2 : t2 = 0
L3 : t3 = 0
L4 : t4 = 4 * t3
L5 : t5 = 4 * t2
L6 : t6 = t5 * M
L7 : t7 = t4 + t6
L8 : t8 = a[t7]
L9 : if t8 <= max goto L11
L10 : t1 = t8
L11 : t3 = t3 + 1
L12 : if t3 < M goto L4
L13 : t'2 = t2 + 1
L14 : if t2 < N goto L3
L15 : max = t1
```

a) 7 and 7 b) 6 and 6

[✓ c)] 7 and 6 d) 6 and 7.

Sol

Block 1 = L1 and L2

Block 2 = L3

Block 3 = L4, L5, L6, L7, L8 and L9

Block 4 = L10

Block 5 = L11 and L12

Block 6 = L13 and L14

Block 7 = L5

=> No of instructions in the largest basic block = 6.

55) Consider the control flow graph shown.

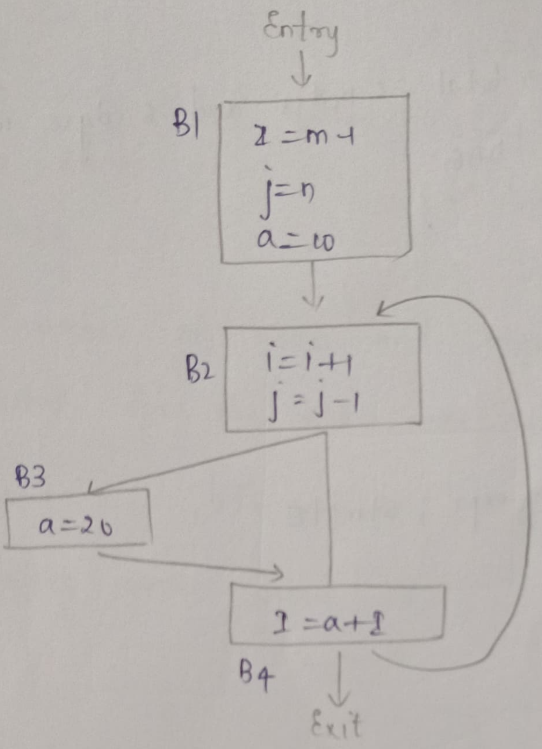
which one of the following choices list the set of live variable at the exit point of each basic block?

a) B1 : { } , B2 : {a} , B3 : {a} , B4 : {a}

b) B1 : {i, j} , B2 : {a} , B3 : {a} , B4 : {i}

c) B1: {a, i, j} , B2: {a, i, j} , B3: {a, i} , B4: {a}

d) B1: {a, i, j} , B2: {a, j} , B3: {a, j} , B4: {a, i, j}.



Sol Option D

Live variables at the end of each basic block are:

B1: {a, i, j}

B2: {a, j}

B3: {a, j}

B4: {a, i, j}.

56) Consider following C code segment:

```
a=b+e;
e=a+1;
d=b+c;
f=d+1;
g=e+f;
```

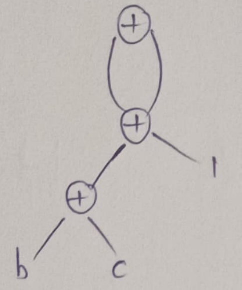
In a compiler, this code segment is represented internally as a directed acyclic graph (DAG).

The no of nodes in the DAG is 6

Sol $g = e+1;$

$\Rightarrow (a+1)+(d+1) \Rightarrow ((b+c)+1)+((b+c)+1)$

Corresponding DAG is:



⇒ total 6 nodes and 6 edges in the DAG

57) Consider the following grammar:

Stmt → if expr then expr else expr ; stmt|ε

expr → term relop term | term

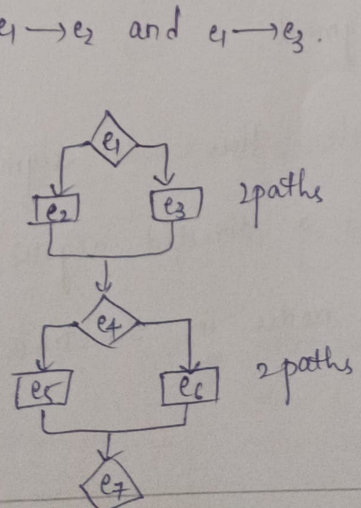
term → id | number

id → a|b|c

number → [0-9]

where relop is a relational operator (eg. <, >, --), o refers to the empty statement and if, then, else are terminals. Consider a program P following the above grammar containing ten if terminals. The number of control flow paths in P is 1024. if e_1 then e_2 else e_3 has 2 control flow paths, $e_1 \to e_2$ and $e_1 \to e_3$.

Sol if e_1 then e_2
else e_3



$2 \times 2 \times$ —— 10 times $= 2^{10}$

$= 1024$ paths.

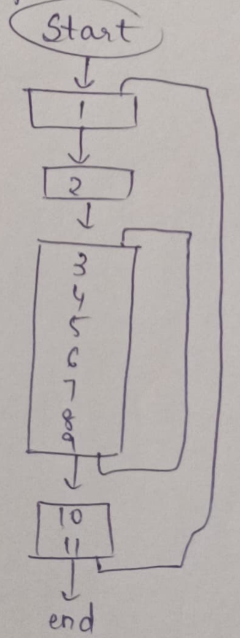
58) Consider the following intermediate code given below:

```
1. i=1
2. j=1
3. t1 = 5*i
4. t2 = t1+j
5. t3 = 4*t2
6. t4 = t3
7. a[t4] = -1
8. j=j+1
9. if j<=5 goto (3)
10. i=i+1
11. if i<5 goto (2)
```

The number of nodes and edges in the control-flow-graph constructed for the above code, respectively are

a) 5 and 7 [✓ b) 6 and 7] c) 5 and 5 d) 7 and 8

Sol Corresponding control graph is



Total, 6 nodes and 7 edges,

So option B is correct.

59) In the context of abstract syntax tree (AST) and control-flow-graph (CFG) which one of the following is true?

a) In both AST and CFG, let node N_2 be the successor of node N_1 in the input program, the code corresponding to N_2 is present after the code corresponding to N1.

b) For any input program, neither AST nor CFG will contain a cycle.

c) The maximum number of successors of a node in an AST and a CFG depends on the input program.

d) Each node in AST and CFG corresponds to at most one statement in the input program.

Sol Option C is correct.

(a) → false. If loop is present then control can move from any statement to any statement.

(b) → false. CFG can have a cycle.

(c) → true.

(d) → false.